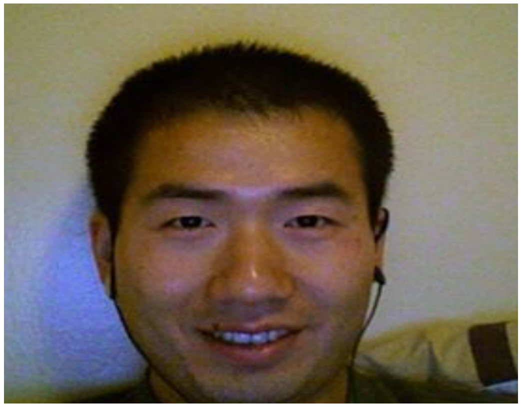
Capability Models and Their Applications in Planning

Yu Zhang, Sarath Sreedharan &

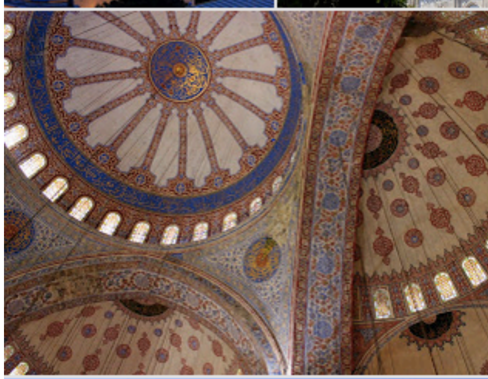
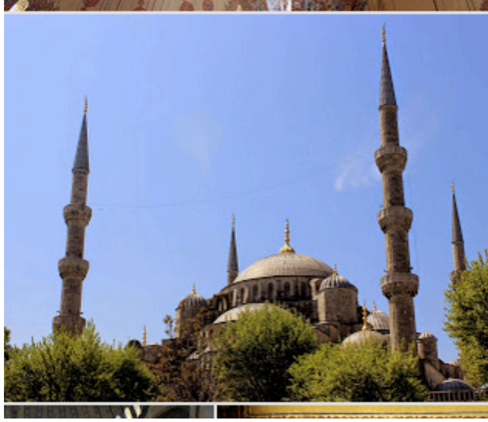
Subbarao Kambhampati

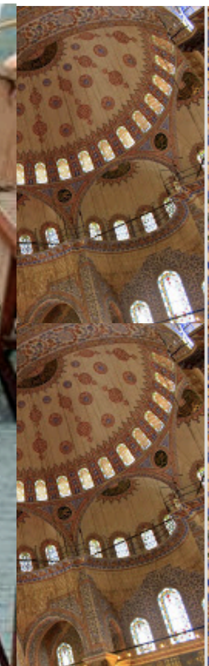
Arizona State University

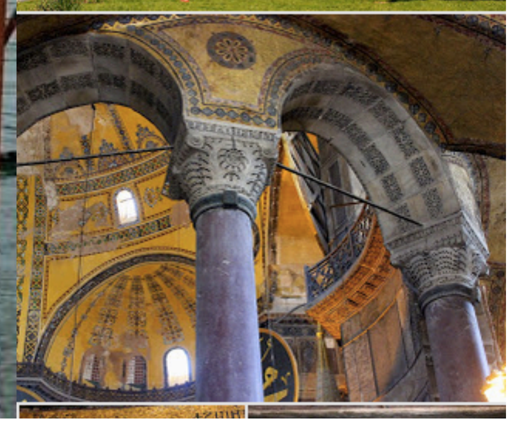




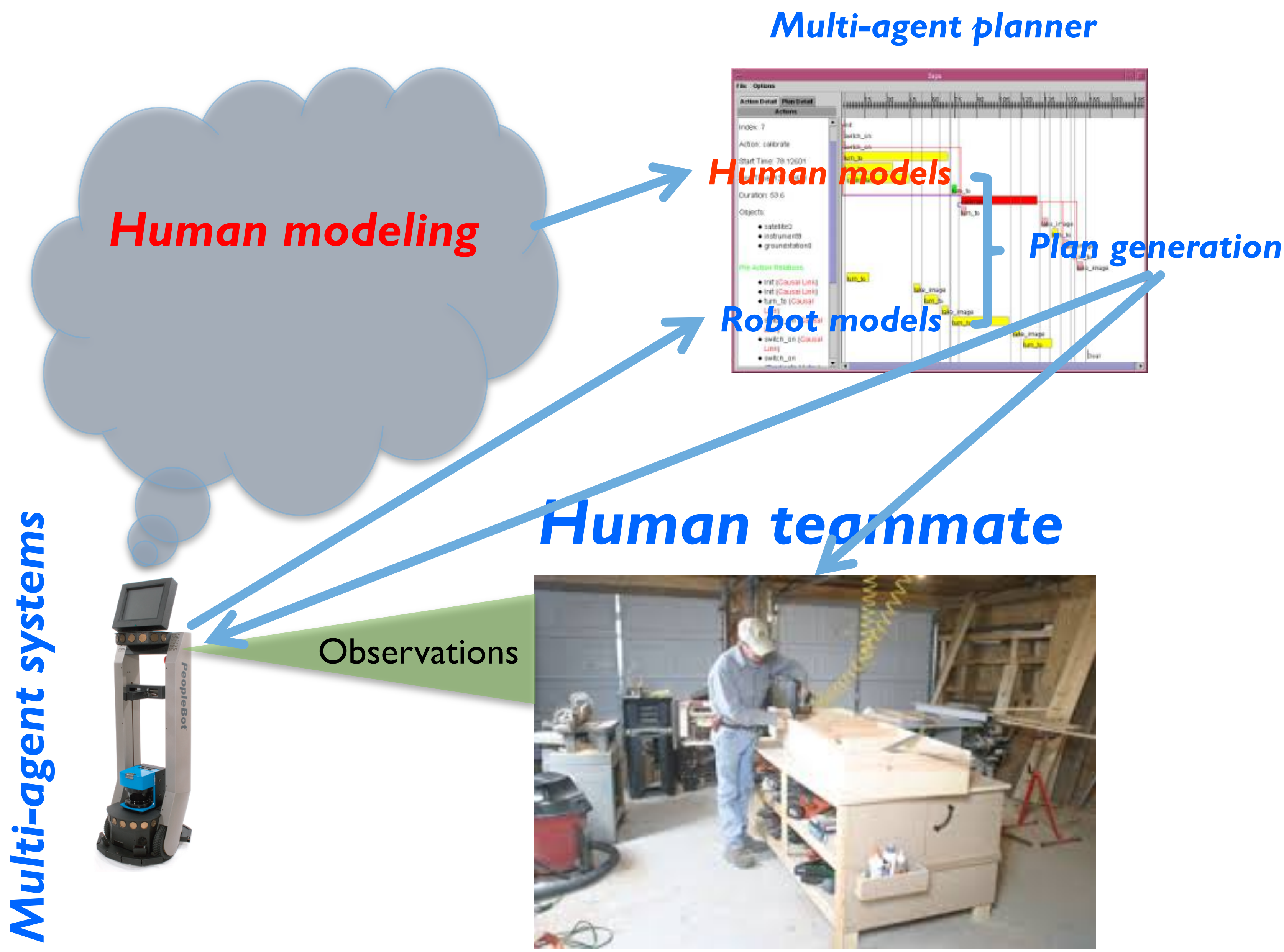




Planning with Humans in the Loop



Human-in-the-Loop Planning & Decision Support

AAAI 2015 Tutorial

rakaposhi.eas.asu.edu/hilp-tutorial

Subbarao Kambhampati
Arizona State University

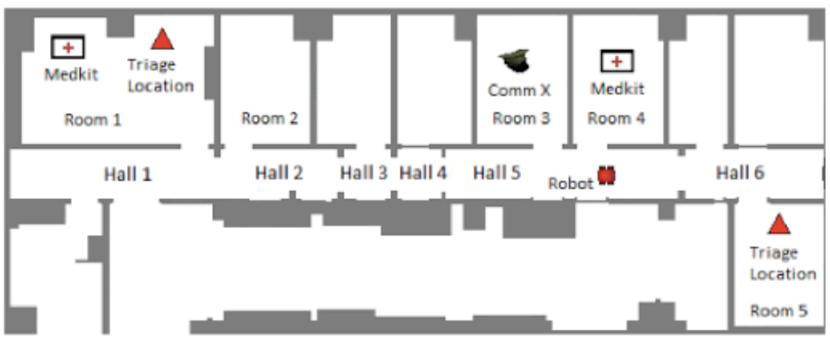
Kartik Talamadupula
IBM T.J. Watson Research Center

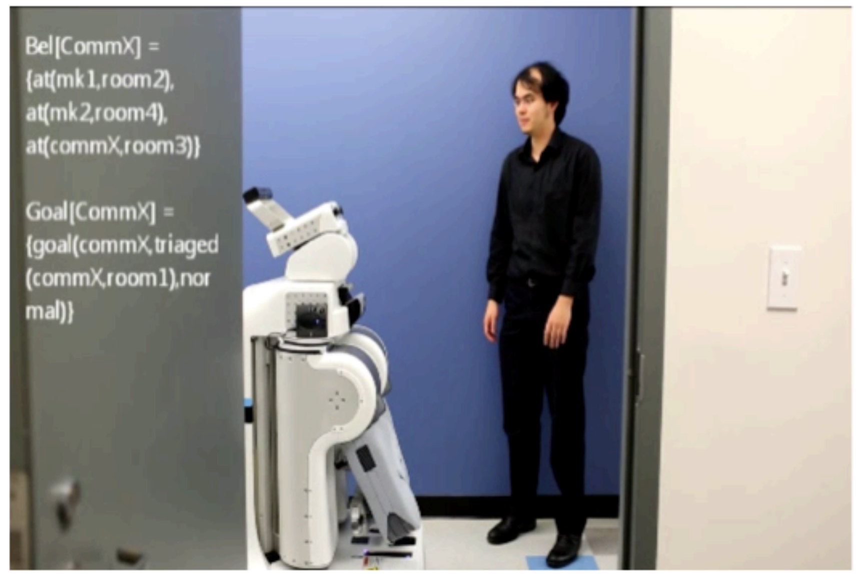
Funding from ONR, ARO and NSF gratefully acknowledged

Interactive Session
Paper TuD2.10

IROS 2014

COORDINATION IN HUMAN-ROBOT TEAMS USING MENTAL MODELING AND PLAN RECOGNITION





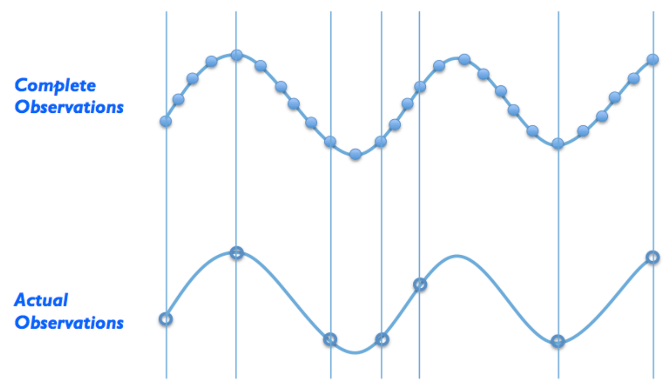
But how do we get the Human Models?

How do we get the Human Models?

- Typically multi-agent planning methods assume all agents use similar models
  - E.g. All agents with STRIPS action models
- Unreasonable to expect similar sorts of action models for the robot and the human..
  - Human models (from the Robot's point of view) are likely to be highly incomplete.
- So how do we represent (and handle) incomplete models of human capabilities?

Challenges in learning Incomplete Human Models

- The temptation is to go with existing action models & introduce incompleteness
  - Atomic: MDP/POMDP
  - Factored: STRIPS, RDDL, HTN etc
    - Example work by Garland&Lesh(2002); Nguyen eal al (2010, 2014)
- While they are fine if someone hand-specifies them, they are much harder to learn, given the kinds of information that is likely to be available.
  - Significant incompleteness in observations
    - Sensor occlusion, noisy observations,
      - [Zhuo & Kambhampati, IJCAI 2013]
  - There may be significant gaps between observations



Observations (partial) with indefinite but bounded gaps

Our Solution: Capability Models

Capability

We start with the "default assumption" that domain models are incomplete

- ***DEFINITION (CAPABILITY)*** **– Given an agent, a capability is a mapping $\boxed{S_\phi} \times \boxed{S_\phi} \rightarrow [0, 1]$, which is an assertion about the probability of the existence of a plan in fewer than or equal to *T* atomic state changes that can connect the two states.**

Partial states

Bound on the gaps between observations

->: denote an atomic state change

{has_water(AG), has_coffee_beans(AG)}
-> {has_boilling_water(AG), has_coffee_beans(AG)}
-> {has_boilling_water(AG), has_ground_coffee_beans(AG)}
-> {has_coffee(AG)}

When T = 2 $\left\{ \begin{array}{l} \text{has\_water(AG) => has\_ground\_coffee\_beans(AG)} \\ \text{has\_boiling\_water(AG) => has\_coffee(AG)...} \end{array} \right.$

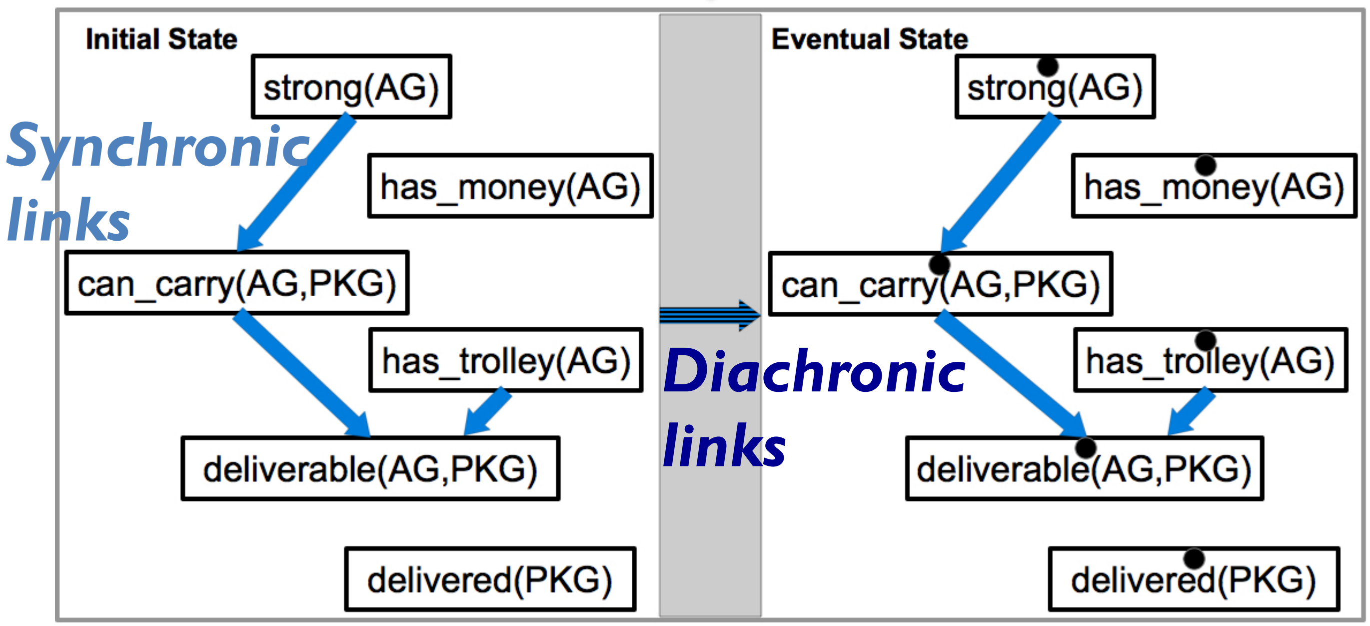
When T = 3 $\left\{ \begin{array}{l} \text{... (including all capabilities when T = 2)} \\ \text{has\_water(AG) => has\_coffee(AG)} \end{array} \right.$

Capability Model

Capability model encodes all capabilities for a given T

(Generalization of 2-TBN model used in RDDL)

T-gap capability model

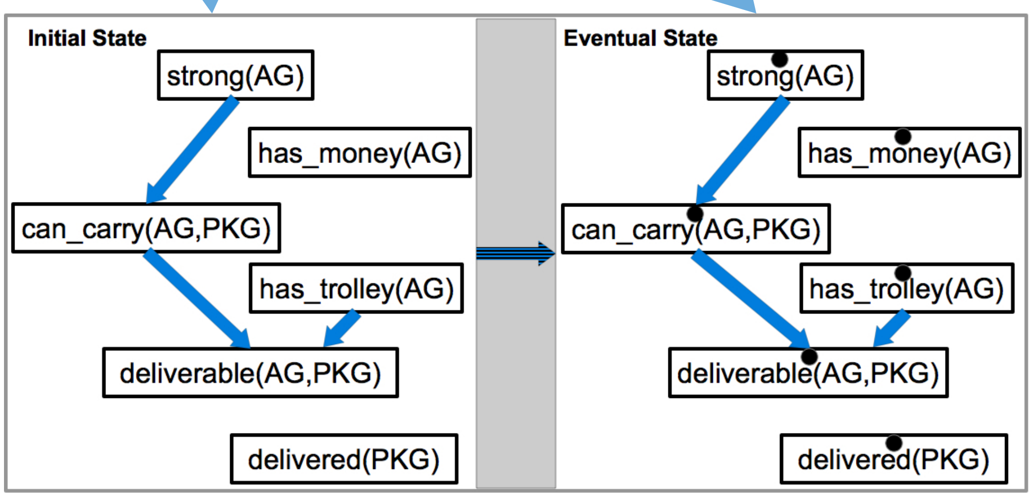


(Imperfect analogy to) HTN Models. A capability can be thought of as an abstract task

Capability Model

DEFINITION 3 (CAPABILITY MODEL). *A capability model of an agent ϕ, as a binomial ABN (G_ϕ, F, ρ), has the following specifications:*

- $V_\phi = X_\phi \cup \dot{X}_\phi$.
- $\forall V_i \in V_\phi$, *the domain of* V_i *is* $D(V_i) = \{$*true, false*$\}$.
- $\forall V_i \in V_\phi$, $F_i = \{F_{i1}, F_{i2}, ...\}$, *and each* F_{ij} *is a root and has a density function* $\rho_{ij}(f_{ij})$ $(0 \leq f_{ij} \leq 1)$. *(For each value* pa_{ij} *of the parents* PA_i*, there is an associated variable* F_{ij}*.)*
- $\forall V_i \in V_\phi$, $P(V_i = true | pa_{ij}, f_{i1}, ..., f_{ij}, ...) = f_{ij}$.

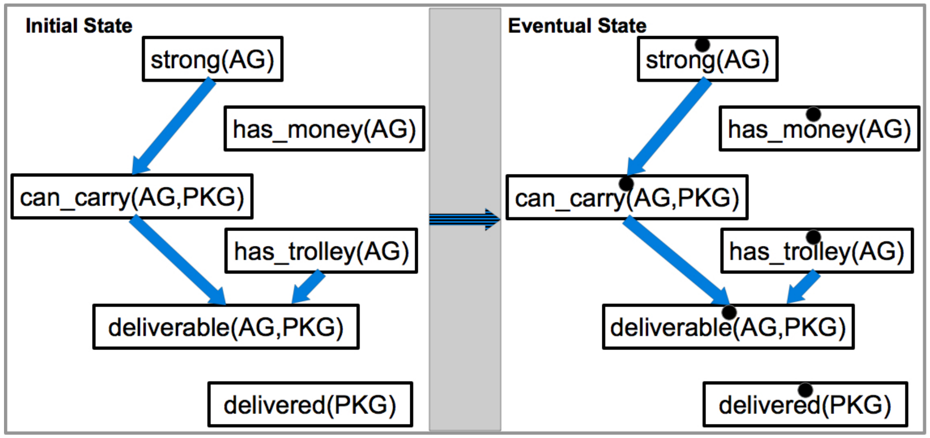


Capability Model & Encoded Capabilities

A capability model encodes the following distributions:

Joint distribution over T

$$P(X_\phi, \dot{X}_\phi) = \int_0^T P(X_\phi, \dot{X}_\phi, t)\, dt$$



T-gap capability model

A capability:

$$P(\dot{X}_\phi = s_E \mid X_\phi = s_I)$$

$s_I \Rightarrow s_E$

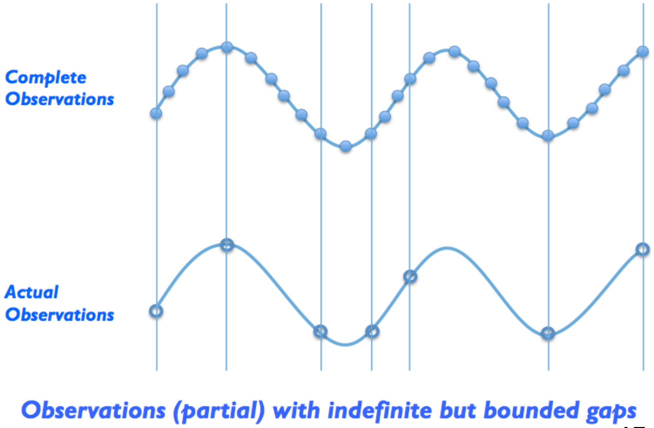
A conditional probability
(specified by a partial initial and eventual state)

Learning Capability Models

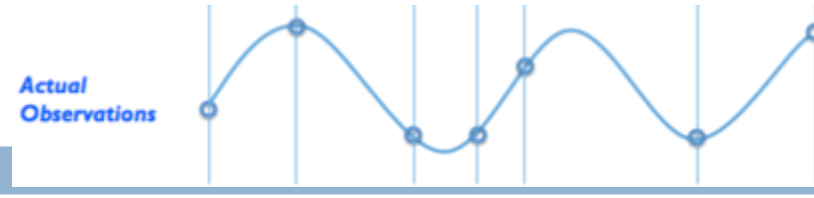
- Learning model structure — Causal relationships (diachronic links); variable correlations (synchronic links)

- Learning model parameters — Conditional probabilities

Learning from (gap-bounded) plan traces



Parameter Learning



We assume that the maximum number of missing state observations between any two observations in the partial plan trace is upper bounded by T

DEFINITION (T-GAP PARTIAL PLAN TRACE). A
T-gap partial plan trace is a partial plan trace in which all $k_{[1,2...]}$ <= *T*

$$\mathcal{T} = \langle s_i, s_{i+k_1}, s_{i+k_2}, \ldots \rangle$$

Learning samples

Apply Bayesian learning (assuming beta distributions):

$$\rho(f_{ij}|D) = beta(f_{ij}; a_{ij} + s_{ij}, b_{ij} + t_{ij})$$

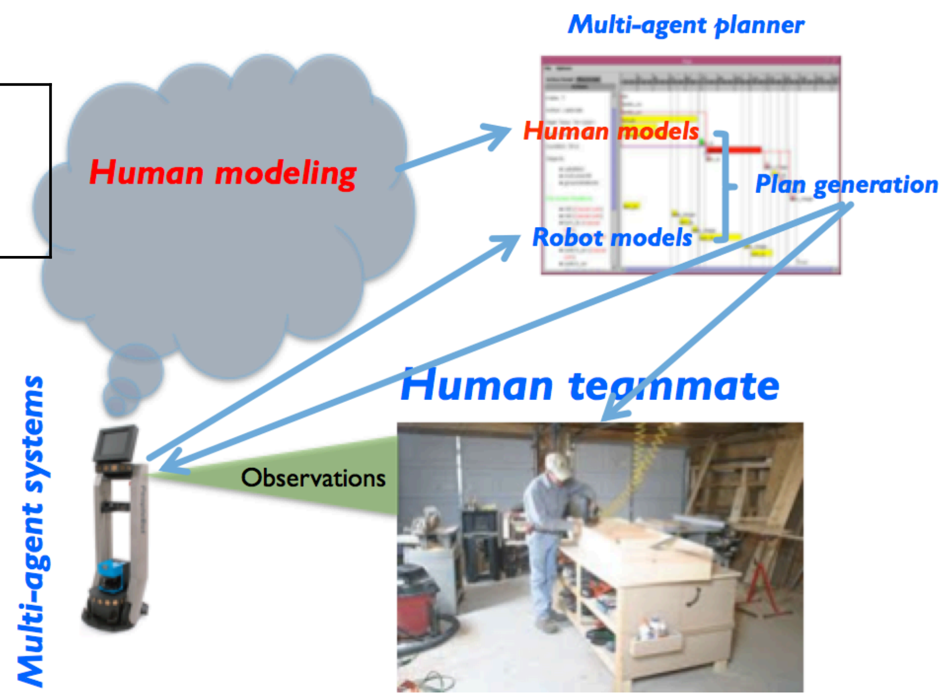
Using Capability Models

Single agent planning

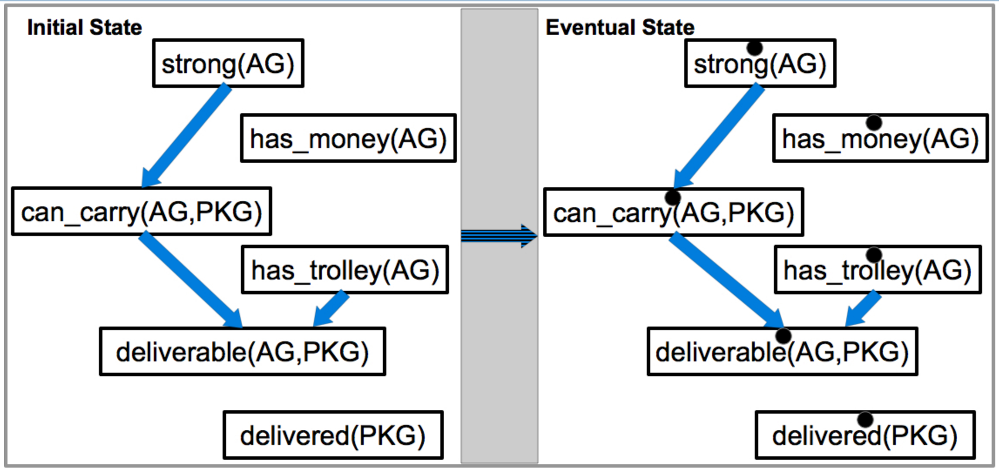
- Robot can reason about whether a human can achieve the task alone

Multi-agent planning (e.g. Robot and Human)

- Robots can reason about a joint plan with humans



Planning with Capability Models



T-gap capability model

- Any planning state is a set of complete states: a belief state

  {(complete state 1), (complete state 2)…}

- Select a capability to apply: $s_I \Rightarrow s_E = P(\dot{X}_\phi = s_E \mid X_\phi = s_I)$
- For each s* in the belief state,
  - Applicable: $s_I \sqsubseteq s^*$

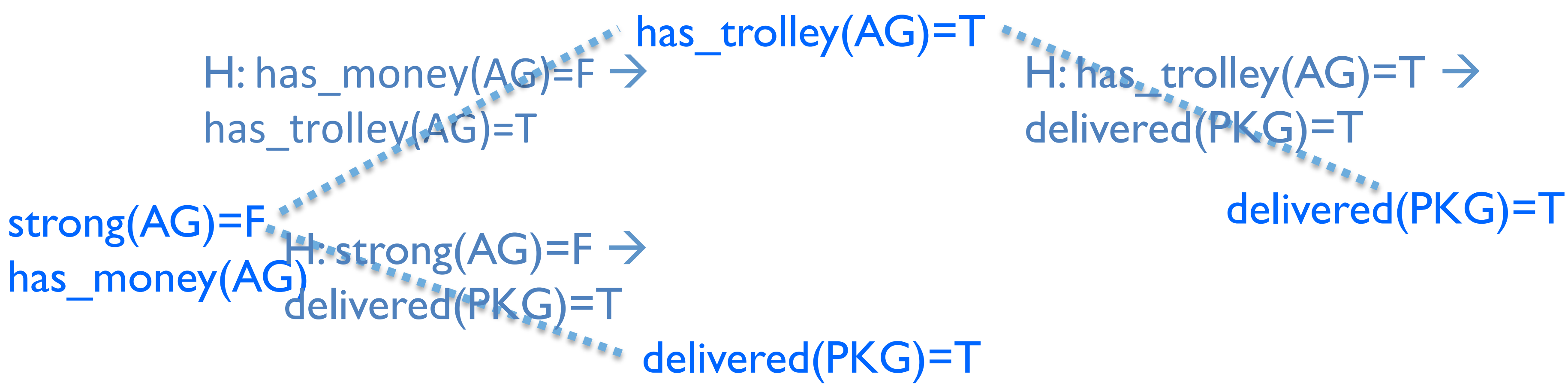
    Success: compute a set of resulting states s, $s_E \sqsubseteq s$,

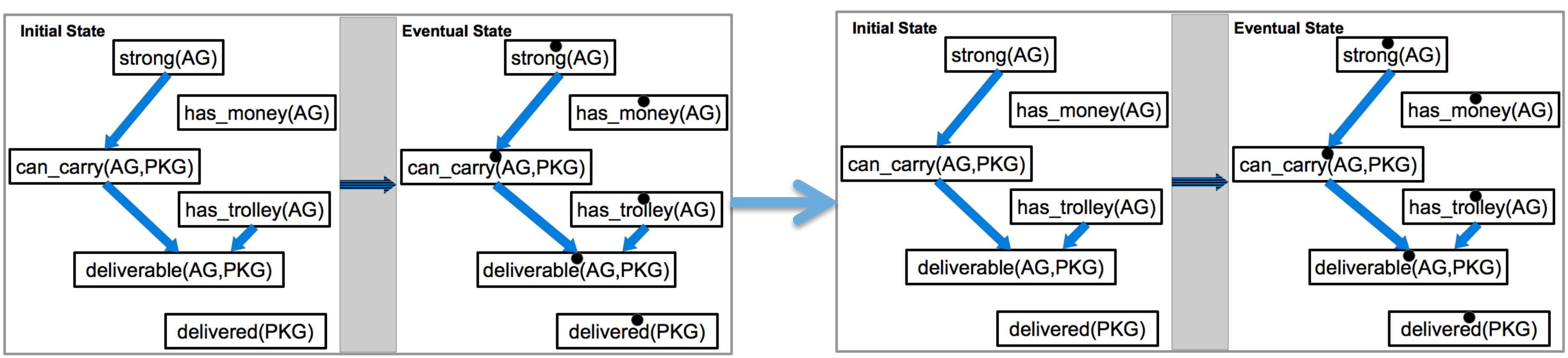
    $$P(s) = \frac{P(s^* \Rightarrow s)}{P(s^* \Rightarrow s_E)} = \frac{P(\dot{X}_\phi = s | X_\phi = s^*)}{P(\dot{X}_\phi = s_E | X_\phi = s^*)}$$

    Failure: no change
  - Inapplicable – no change to s*

$\sum_{s \in S} P(s) = 1$

Single-agent Planning





Unrolling of 2-gap capability model

Single Agent Planning Heuristic

Assumptions:

$$P(s_I \Rightarrow s_E) \geq P(s'_I \Rightarrow s_E)(T(s'_I) \subseteq T(s_I) \wedge F(s_I) \subseteq F(s'_I))$$

$$P(s_I \Rightarrow s_E) \geq P(s_I \Rightarrow s'_E)(T(s_E) \subseteq T(s'_E) \wedge F(s_E) \subseteq F(s'_E))$$

A* heuristic

Given any state s* in belief state b(S):

Compute f(s*) = g(s*) + h(s*)

g(s*) = cost of capabilities in the plan prefix

The cost of a capability is taken as the negative log of the associated probability

h(s*) = $\underset{v \in G_s, s_{\neg v}}{\operatorname{argmax}} -\log P(s_{\neg v} \Rightarrow \{v = \mathit{true}\})$

- G_s is the set of variables that still need to be made true
- $S_{\sim v}$ is a complete state with all variables being TRUE except for v
- {v = true} is a partial state in which v is true

$$h(\hat{b}(\mathcal{S})) = \sum_{s \in \mathcal{S}} P(s) \cdot h(s)$$

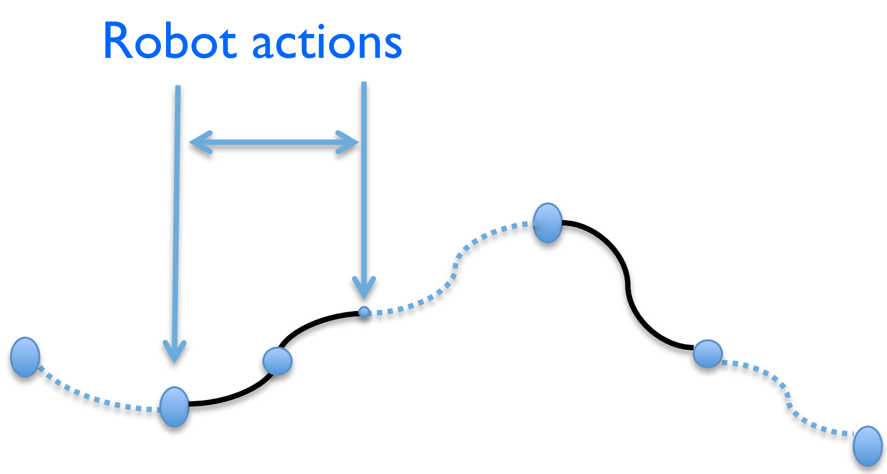
Multi-agent Planning Problem

- For robotic, agents, we assume STRIPS action models
  - Apply action model on any complete state in the belief state is straightforward
- For human agents, we assume capability models

DEFINITION 8. *Given a set of robots* $R = \{r\}$, *a set of human agents* $\Phi = \{\phi\}$, *and a set of typed objects* O, *a multi-agent planning problem with mixed models is given by a tuple* $\Pi = \langle \Phi, R, b(\mathcal{I}), G, \rho \rangle$, *where:*

- *Each* $r \in R$ *is associated with a set of actions* $A(r)$ *that are instantiated from* $\mathcal{O}$ *and* O, *which* $r \in R$ *can perform; each action may not always succeed when executed and hence is associated with a cost.*
- *Each* $\phi \in \Phi$ *is associated with a capability model* $G_\phi = \langle V_\phi, E_\phi \rangle$, *in which* $V_\phi = X_\phi \cup \dot{X}_\phi$. $X_\phi \subseteq X$, *in which* X_ϕ *represents the state variables of the world and agent* ϕ *and* X *represents the joint set of state variables of all agents.*

Planning with mixed models!



Multi-agent Planning

has_trolley(AG)=T

H: has_trolley(AG)=T → delivered(PKG)=T

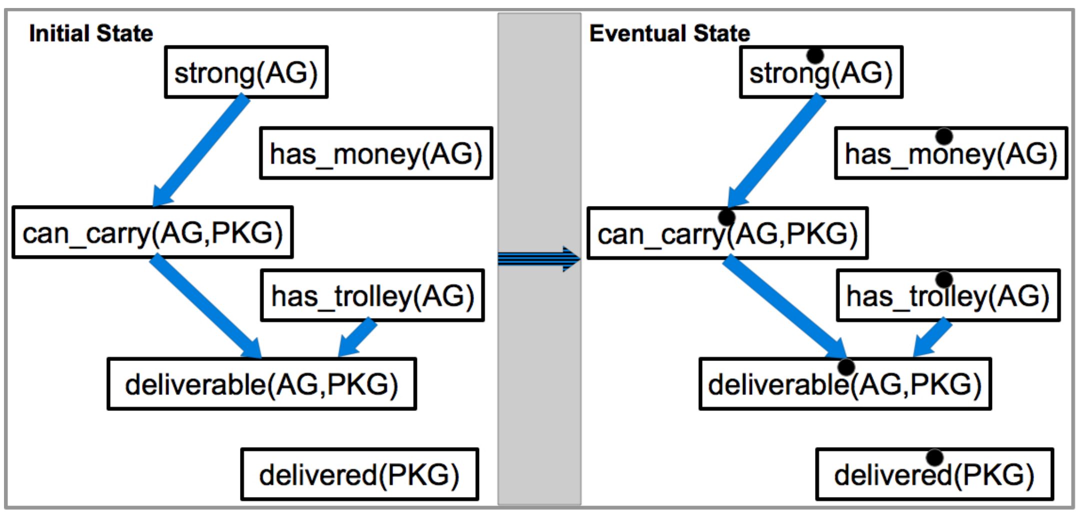
R: Move a trolley
Inside the truck

delivered(PKG)=T

strong(AG)=F

H: strong(AG)=F → delivered(PKG)=T

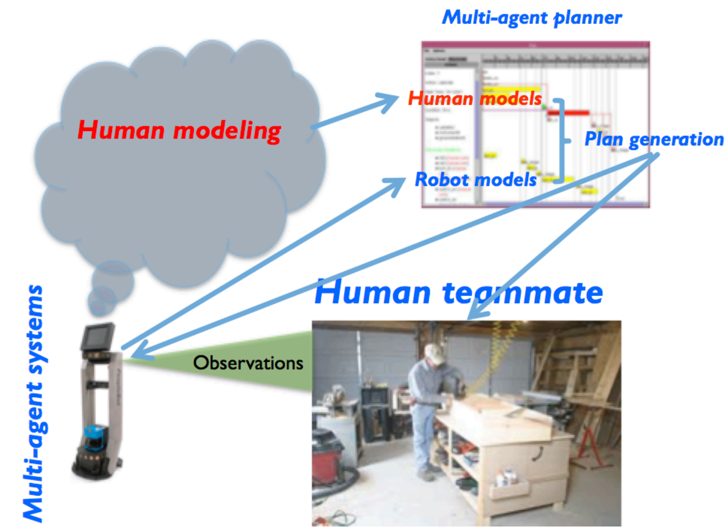
delivered(PKG)=T

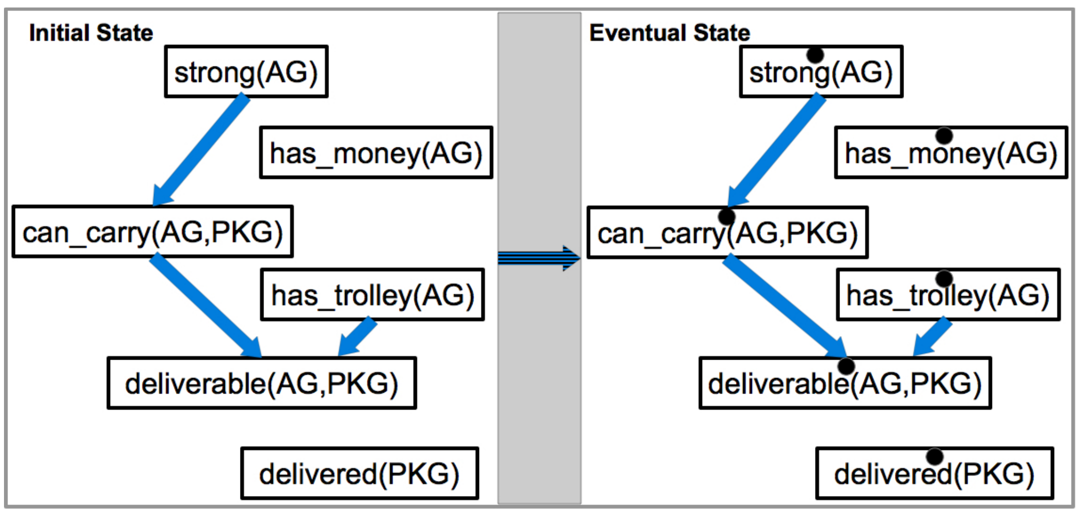


2-gap capability model

Conclusions

- **Introduced capability models for human modeling**
- **Discussed learning and planning with capability models**
- **Preliminary evaluation in the paper..**





T-gap capability model

Start with the "default assumption" of incomplete domains

- ***Learn from observations with indefinite but bounded gaps***
- ***Non-angelic uncertainty***

C-plan